ORACLE
ZFS STORAGE APPLIANCE

Performance Tuning the Oracle ZFS Storage Appliance for Microsoft Exchange 2013

ORACLE WHITE PAPER | MARCH 2016

ORACLE

Table of Contents

Introduction

The Oracle ZFS Storage Appliance is formed by a combination of advanced hardware and software architecture with the purpose of offering a multiprotocol storage subsystem. The unified storage model enables you to simultaneously run a variety of application workloads while benefitting from advanced data services. The first-class performance characteristics of the Oracle ZFS Storage Appliance are illustrated by the results of industry-standard benchmarks like SPC-1, SPC-2 and SPECsfs.

As a unified storage solution, the Oracle ZFS Storage Appliance provides an excellent base for building Windows solutions with iSCSI or Fibre Channel block storage for Microsoft Exchange Server, for example, and NFS or SMB file storage for Microsoft Windows client access.

This paper discusses the performance tuning available as part of the infrastructure of the Oracle ZFS Storage Appliance to support Microsoft Exchange Server in various scenarios.

The test procedure used throughout this document is based on the Microsoft Exchange Solution Reference Program tools – JETstress – which is designed to simulate Microsoft Exchange Server disk I/O load and to produce performance statistics on the resulting operations.

Performance Tuning

The goal of performance tuning is the improvement of system or application speed. The application can usually be considered the top level of the hierarchy of components involved in tuning, and is usually the area of expertise of the application developers. For the purpose of this document, it is assumed that the application is immutable. The focus is on the lower layers – the file system and the storage subsystem itself.

The art of performance tuning revolves around knowing what variables there are to modify and what effect the modification will have to the system as a whole, as tuning a variable may cause undesired effects at another level.

The application traditionally sits on top of the file system where it stores persistent and, optionally, temporary data, state information, and metadata. File system performance is a key criterium in the performance of the application as perceived by the user.

The file system recommended for use with Microsoft Exchange Server is NTFS.

Tuning the File System

There are not many parameters that can be changed in an NTFS file system when installing with Microsoft Exchange. Parameter changes require file system reformatting that can result in the loss of all contained data unless it is backed up elsewhere and restored to the newly formatted file system. Therefore, it is very important that the formatting parameters are set properly when the file system is initially created.

Microsoft recommends a 64 KB allocation unit size, which defines the smallest file size that will be allocated when a file is created. For example, a 1-byte file would take up 64 KB of disk space. An optimization built into NTFS does allow very small files to be stored in the Master File Table (MFT) without taking up any additional units, but the file sizes involved with Microsoft Exchange preclude this particular automatic optimization.

NTFS allocates files in multiples of the allocation unit size; when a file grows over the allocated size, another unit is allocated to the file. The allocation unit size provides hints on the IO blocksize that Windows Server will use to satisfy application requests: the larger the allocation unit size, the larger the IO blocksize.

This is consistent with the large file sizes of the .EDB files employed by Microsoft Exchange Server and is also the largest allocation unit size that can be selected. Taking this approach reduces the number of allocation operations carried out as the 1 MB log files are created and filled and also as the database files are created and grow.

The volatile nature of Microsoft Exchange database files with their dynamic growth and shrinkage can lead to fragmentation of the file system. A series of records in the database files which are logically concurrent may spread over multiple blocks in the underlying storage system.

By default, the format operation for a volume selects the following allocation unit size for Windows Server 2008 onwards:

TABLE 1. NTFS DEFAULT ALLOCATION SIZES

VOLUME SIZE	ALLOCATION UNIT SIZE
7 MB – 16 TB	4 KB
16 TB – 32 TB	8 KB
32 TB – 64 TB	16 K
64 TB – 128 TB	32 K
128 TB – 256 TB	64 KB

Microsoft Windows Server does not support NTFS volumes larger than 256 TB.

It is not easy, using standard Windows Server tools, to find out the allocation unit setting in existing infrastructure. Use the command line tool `fsutil` to analyze the NTFS file system details. Figure 1 shows an example.

See further details about `fsutil` at https://technet.microsoft.com/en-us/library/cc753059.aspx.

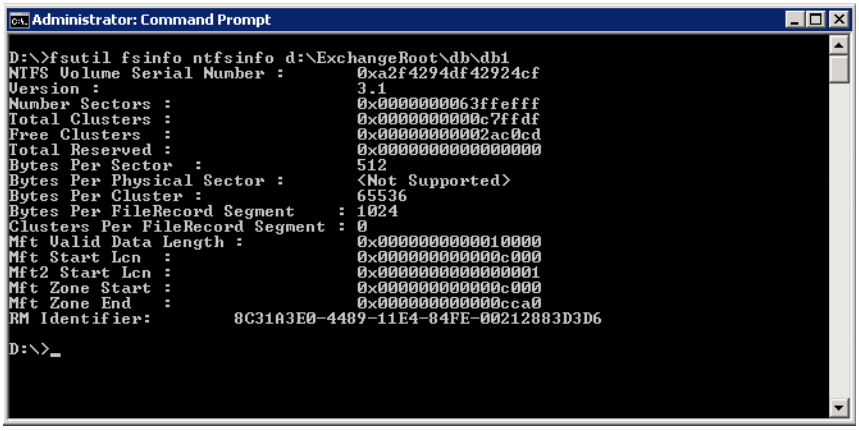

Figure 1. Using the `fsutil` command for NTFS file system information

The important figure in the output of `fsutil` is the Bytes Per Cluster entry which shows the allocation unit size. As shown in the example, the file system mounted on `D:\ExchangeRoot\db\db1` has a Cluster (allocation unit) size of 65536, which represents 64 KB.

The file system makes use of the facilities offered by the storage subsystem – in this case, the Oracle ZFS Storage Appliance. A number of storage optimization and services supported by the Oracle ZFS Storage Appliance can improve the overall performance of Microsoft Exchange deployments.

Tuning the Oracle ZFS Storage Appliance

Microsoft Exchange Server attempts to write blocks as efficiently as possible, depending on the blocksize of the provisioned storage and also the volume data type: logs or database. Log volumes are written in approximately 8 K chunks, or lower if the storage device has a restriction on the maximum block size.

However, during log replication, the average size of the reads from the log volumes appears to be around 225 kbytes. Contrasting with the database volumes, reads are approximately 32 K and writes just over 36 K. This presents a challenge to the tuning process, as the blocksizes vary greatly for different operations on the same volumes – particularly log volumes.

As a result, there is no “one size fits all” rule that can be applied to the storage, and the appropriate parameter values depend on the priority to which the Exchange Server needs to respond. For instance, a volatile system with many incoming new emails will have a different requirement for the sizing of the underlying volume blocksizes than an archive system, which is predominantly read-only.

You must also consider the day-to-day housekeeping tasks undertaken by Microsoft Exchange Server as they can impact the performance perceived by the Exchange users. And to further complicate the mix, backup performance also must be considered, given the way Exchange Server backups operate, and particularly when backups occur during normal operation hours.

Log Volume Characteristics and Challenges

Microsoft Exchange Server log volumes are used to track changes to be applied to the database. When a message is received, sent, deleted or changed, the transaction is recorded in the log files and in memory until the transaction

is committed to the database files. Once the change has successfully been applied to the database files, the transaction is marked as actioned and acknowledged to the user. The performance of log volumes is arguably the most user-visible component within Microsoft Exchange, as all transactions to modify data in any way (update, delete, modify) are processed through the log volumes.

The log files are not purged during normal operation but only after the Microsoft Exchange Server data is backed up, so the number of these 1 MB files can grow quite large on a highly active server.

Log files are critical to recovery options for a failed Exchange Server, where these failures can rank from minor crash up to data loss. The log files are not purged in order to ensure that the ability to roll forward or backwards is available to the administrator in the event of a catastrophic failure.

The access profile for log files is reasonably sequential as transactions are added to the log files but with a random element as the transactions are marked as 'actioned.'

Transaction logs can (and arguably should) be replicated automatically by combining Microsoft Exchange Servers into a Database Availability Group (DAG). DAGs provide a replication domain boundary to ensure that the loss of a single server (and associated database and log files) does not affect the availability of the data. They are provided by the inclusion of Microsoft Exchange Active Manager. For the purposes of this document, DAGs are only considered as additional IO requests to the database and log volumes. For further details regarding DAGs, see http://technet.microsoft.com/en-gb/library/dd979799(v=exchg.150).aspx

As indicated previously, the IO characteristics of Microsoft Exchange Server transaction logs are more complex than those of the database files. Log files grow to a maximum of 1 GB and a new transaction log file is then created and populated.

Database Volumes

The IO profile of requests to the Microsoft Exchange database volumes is more consistent than that of the log volumes. Microsoft Exchange Solution Review Program (ESRP) statistics show averages of 33,100 bytes read between instances of the database files across all the tests shown and 37,100 bytes written between instances.

Unfortunately the statistics provided by ESRP only show mean and do not give any indication of standard deviation. Using the Oracle ZFS Storage Appliance Advanced Analytics feature, more information regarding the spread of the request sizes shows that the standard deviation is relatively high. With a large spread over IO sizes, traditional storage subsystem design would address these allocations through compromise, hopefully hitting the highest priority operation blocksize. The adaptive caching algorithms within the Oracle ZFS Storage Appliance can smooth over the variance in the request size.

Database files are also replicated as part of the DAG scheme provided by Microsoft Active Manager as discussed in the Log Volumes section.

Performance Test Procedure

Due to the terminology conflict between Exchange logs and Oracle ZFS Storage Appliance log devices, the term "log volume" will be used to specify the Exchange data type and "log device" to specify the Oracle ZFS Storage Appliance SSD log device.

The performance tests were carried out using a single Oracle Sun Server X3-2 running Microsoft Windows Server 2008 R2 with all the recommended, security and optional updates applied up to the date of testing.

The Oracle Sun Server X3-2 was configured as follows:

TABLE 2. EXAMPLE ORACLE SUN SERVER X3-2 CONFIGURATION FOR PERFORMANCE TESTS

COMPONENT	SPECIFICATION
Processor	2 x Intel Xeon E5-2690 @ 2.9GHz
Cores per Processor	8
Main Memory	72.0 GB
Operating Environment	Windows Server 2008 R2 Enterprise – Service Pack 1
Storage Connectivity	Fibre Channel SAN

LUNs were presented from the Oracle ZFS Storage ZS3-2 that was configured as a single (non-clustered) head with three disk shelves. During the test, only one shelf-worth of drives was used – 20 HDDs and 4 SSD log devices. Internally, the Oracle ZFS Storage ZS3-2 had 2 x 1.6-TB Cache SSDs and 2 x 83 8-GB system HDDs.

The storage was provisioned as two RAID-1 pools with one for database volumes and the other for log volumes as per ESRP2013 rules. All presented LUNs were configured with varying levels of compression (as described later), no encryption, and as full provisioned volumes. Caching was set for both data and metadata and write-bias is set for latency.

The following table describes the storage configuration:

TABLE 3. STORAGE CONFIGURATION FOR TESTING EXAMPLE

COMPONENT	HDD CONFIGURATION	CACHE CONFIGURATION	SSD CONFIGURATION
Log Device Pool	RAID 10, 4 x 3TB SATA 7200RPM HDD	1x 1.6TB SSD	1 x 73GB SSD
Database Device Pool	RAID 10, 14 x 3TB SATA 7200RPM HDD Hot spare 2 x 3TB SATA 72000RPM HDD	1 x 1.6TB SSD	3 x 73GB SSD Stripe

LUNs were provisioned as follows:

TABLE 4. LUN PROVISIONING FOR PERFORMANCE TESTING EXAMPLE

LUN TYPE	NUMBER	SIZE	POOL
Log Volume	4	150 GB	Log Device Pool
Database Volume	4	800 GB	Database Device Pool

Connectivity between the Oracle ZFS Storage Appliance ZS3-2 and the Oracle Sun Server X3-2 was over a Fibre-Channel network controlled by Brocade 8Gb 300C Fibre-Channel SAN switches.

In order to gauge Microsoft Exchange Performance, the Exchange Solution Reference Program (ESRP) 2013 version test suite was installed along with the required JET (Microsoft JET Database Engine) dynamic link libraries from the Microsoft Exchange Server 2013 installation media.

The following are the parameters used for the statistics-gathering runs.

TABLE 5. TEST PARAMETERS AND RELATED SETTINGS AND SCOPE

TEST PARAMETER	SETTINGS/SCOPE
Test Category	Disk Subsystem Throughput
Percentage of Storage Capacity	100
Thread Count	50
Test Type	Performance
Background Maintenance	Enabled
Test Duration	2 hours
Number of Databases	4
Number of Copies per Database	2
Database Source	LUNs reinitialized and database restored from a backup copy

These test parameters were chosen to stress the storage subsystem and are not representative of a production Microsoft Exchange deployment.

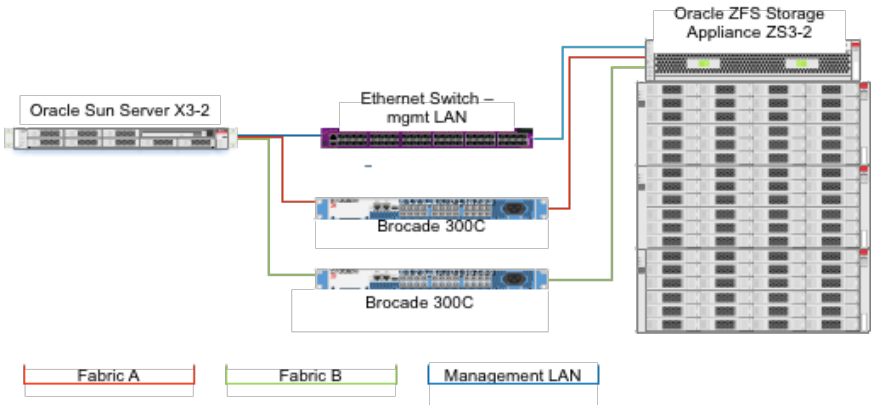

Figure 2. Oracle ZFS Storage ZS3-2 test configuration

Latency in Exchange Volume Types

Within a storage infrastructure, latency is unavoidable and can occur in many operational stages: when requests are transferred over the storage area network which is located a finite distance from the host bus adapter over copper or fibre, when the request is received at the storage system and is translated into individual cache and disk operations, which in turn require the controllers to wait for data to be retrieved or written, and the subsequent acknowledgement

is then returned to the application back over the storage network infrastructure. All these operations cost time and cumulatively form the request latency.

Performance tuning can assist in reducing the latency to the minimum possible (there will always be some latency) by analyzing the behavior of the storage systems under different IO request patterns and sizes.

Log Volume Blocksize for Latency

While Microsoft recommends a 64 KB blocksize for LUNs presented to Exchange Server, testing served to investigate how this recommendation holds up against the Oracle ZFS Storage Appliance with its advanced dynamic performance management features.

With the Oracle ZFS Storage Appliance, blocksizes for volumes can range from 1 KB to 1 MB in standard doubling progression.

The following figure shows the performance characteristics of Exchange 2013 ESRP running on different blocksizes. The generated statistics are based on the server running the tests as opposed to those indicated by the Oracle ZFS Storage Appliance advanced analytics. This approach was taken to allow the inclusion of infrastructure latency, and as more representative of the user experience.

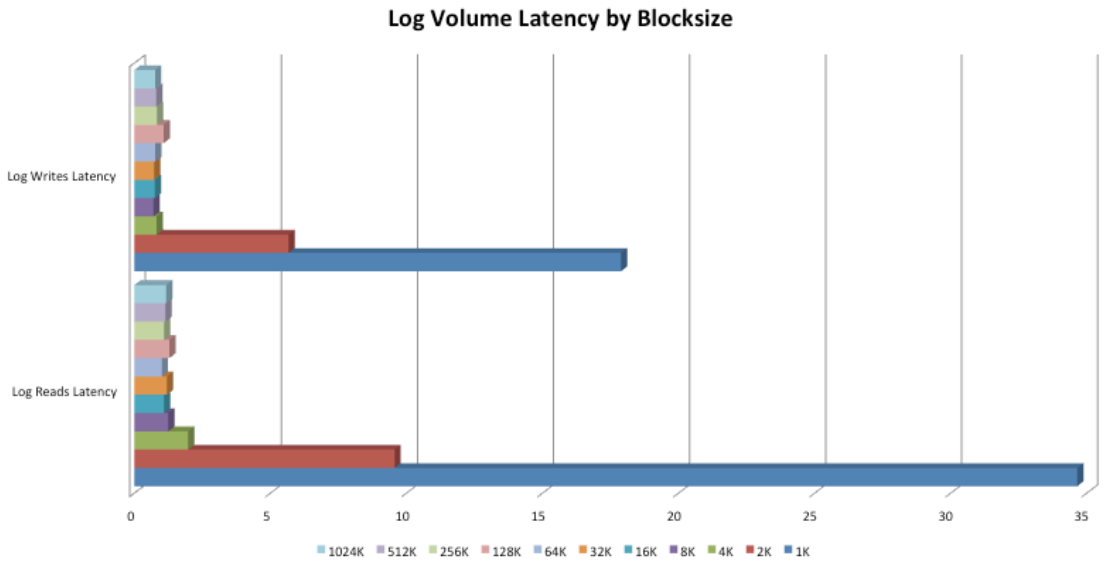

Figure 3. Log volume latency (ms) by blocksize

There is a clear performance impact in the lower end of the blocksize range. So discarding the 1 K and 2 K results to get a clearer picture of the relative performance results, the following figure shows more detail in the 4 KB to 1024 KB range.

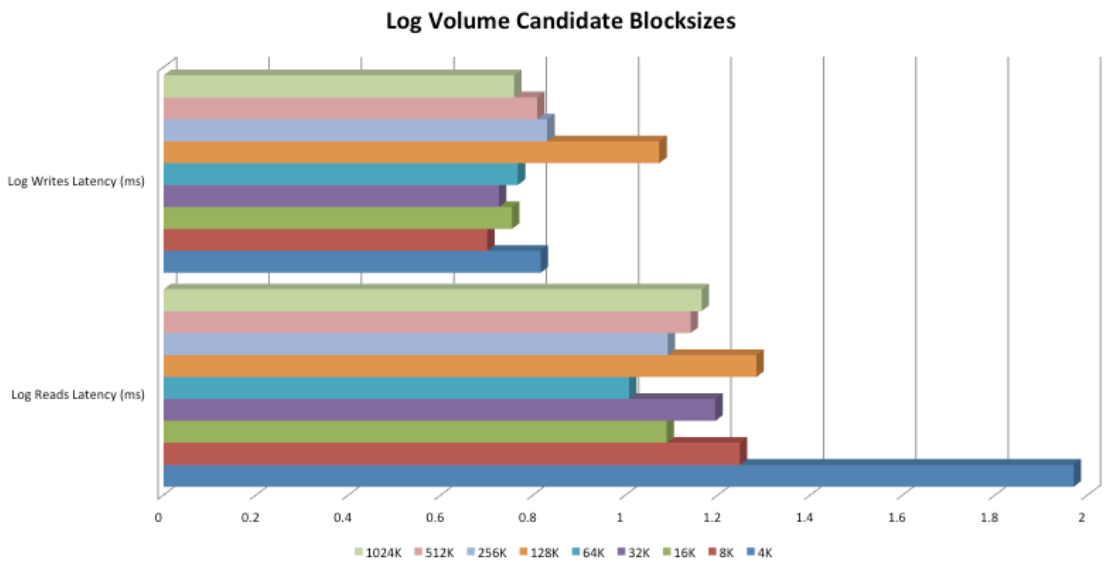

Figure 4. Candidate blocksizes for log volumes

As figure 4 illustrates, the lowest latency results are shown for writes in the 32 K blocksize and for reads in the 64 K blocksize entries. In practice, the difference for writes is only 0.03ms, or 7 percent, and for reads 0.14ms, or 13 percent.

As stated earlier, blocksize choice will be a compromise between reads and writes. This 64 KB blocks option shows the least impact or difference between reads and writes incurred.

Database Volume Blocksize for Latency

As with log volumes, the performance measured by individual transaction latency can vary with the underlying volume blocksize. In order to determine the optimal blocksize for a typical Exchange Server as modeled by JETStress, the database volumes were configured in the standard doubling blocksize from 1 K to 1 M, in the same manner as log volumes.

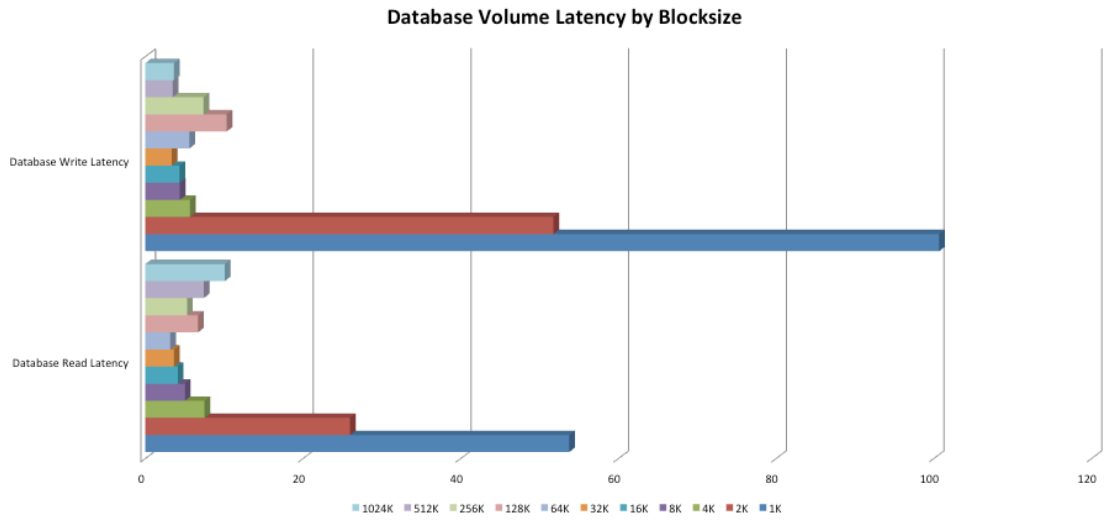

Figure 5. Database volume latency by blocksize

Figure 5 shows that, similarly to the log volume results, there is a marked performance impact in database volume latency when choosing a 1 KB or 2 KB blocksize. Removing these values from the equation gives the following results:

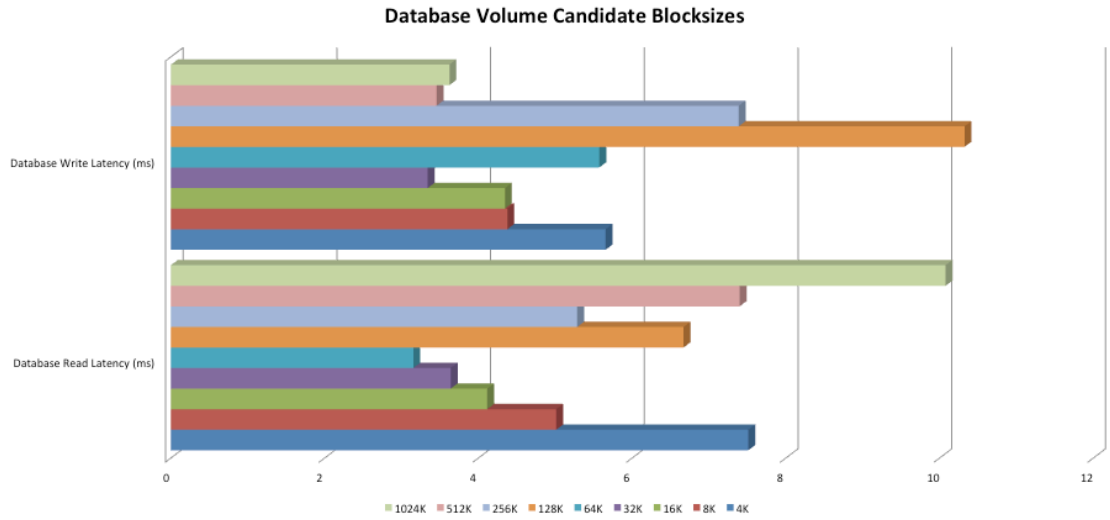

Figure 6. Candidate blocksizes for database volumes

Clearly from the preceding figure, the variance in blocksizes impacts the performance more than the log volumes from the previous section. Read latency is smallest for 32 KB blocks but lowest for writes at 64 KB. Once again, there will have to be a compromise as there is no single blocksize match for overall best performance.

In this case, the smallest difference is in the 32 KB block entry – the write performance is best at 32 KB and the average read performance varies only by 0.51ms between 32 KB and 64 KB – which accounts for an increase of approximately 15 percent.

So in the general-use case where the latency of each transaction is important, 32 KB appears to be the best choice for database volume blocksize when hosted on the Oracle ZFS Storage Appliance.

Selecting Volume Blocksize for Throughput

Looking at another aspect of performance, another key factor is the total throughput of the system – perhaps at the sacrifice of individual transaction latency. In a very busy system, it may be tolerable to have a 10ms average transaction time when it means an overall system throughput increase of 25 percent. In a perfect world, the highest throughput and lowest latency would be at the same blocksize, but as the following sections will illustrate, that is not necessarily the case.

Log Volume Blocksize for Throughput

The average throughput per second in the statistics provided by ESRP2013 shows the following:

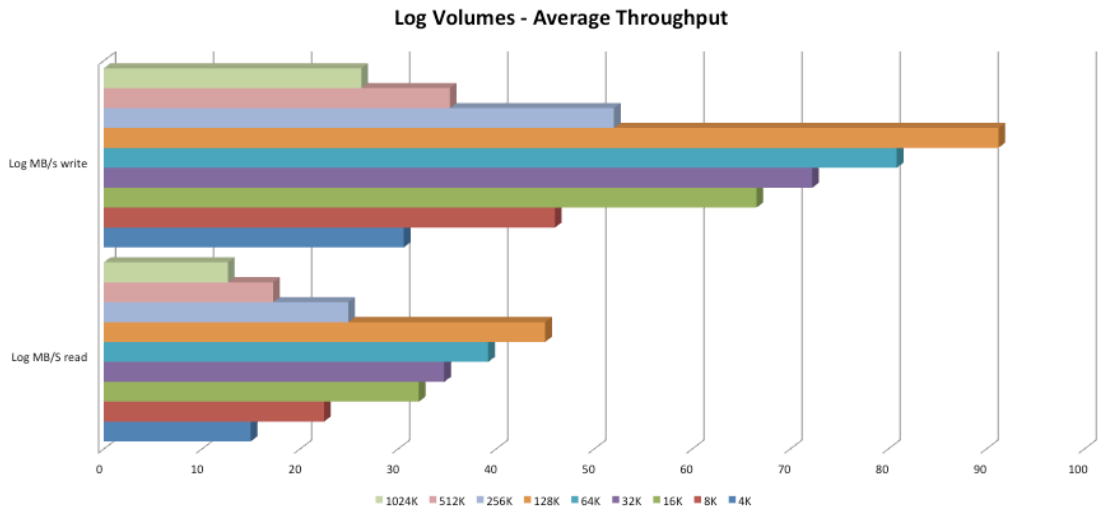

Figure 7. Log volumes -– average throughput (MB/s)

As shown in figure 7, the highest average throughput achieved for the log volumes corresponds to a blocksize of 128 KB, where the lowest latency from figure 3 was at 64 KB. In practice, the latency value is still more relevant unless the log volumes are 100 percent busy all the time, in which case, throughput has more relevance.

When running ESRP tests, the size of the queue of log volume storage requests never gave any indication that the log volumes were causing a bottleneck in the system.

Database Volume Blocksize for Throughput

Focusing on the database volumes, the ESRP2013 statistics were used to find the average throughput, shown in Figure 8.

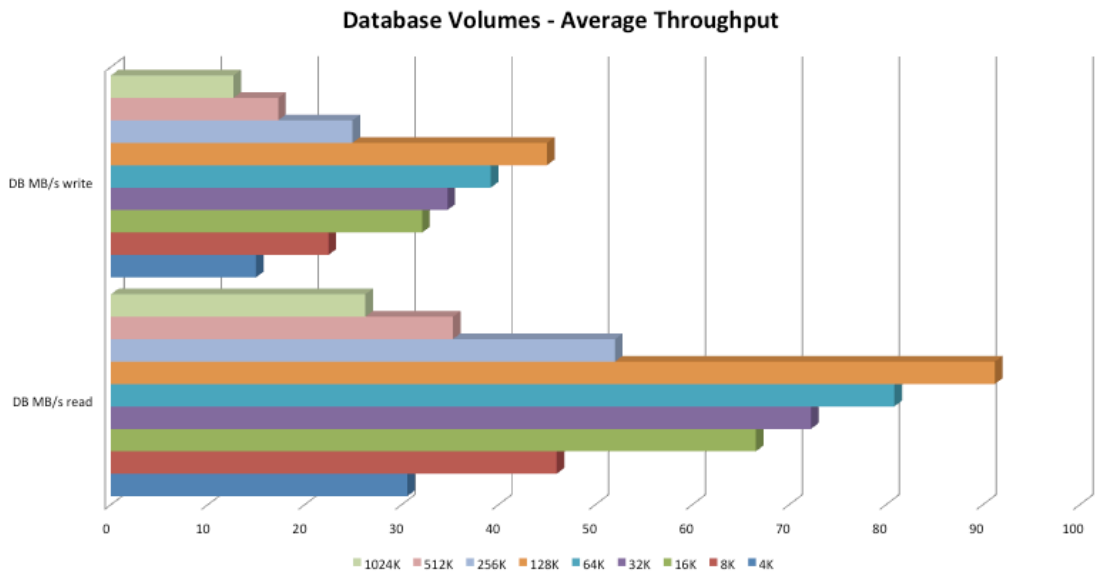

Figure 8. Database volumes – average throughput (MB/s)

Once again, the peak throughput for these volumes occurs at the 128 KB blocksize point. With a difference of 12.7 percent from the next nearest performing blocksize (64 KB) database read throughput may provide a compelling reason to opt for this blocksize, particularly in Microsoft Exchange deployments with heavy search operations. Similarly, the database volumes write throughput with a 128 KB blocksize exceeds the 64 KB entry by 14.1 percent. This is really only relevant when the database volumes are busy – the cost of individual transaction time against the throughput achieved may be an important criteria for a particular deployment.

ZFS Database Blocksize

The database blocksize is used to provide a suggested blocksize for files within a file system. As this value refers only to file systems presented from the Oracle ZFS Storage Appliance, this variable has no relevance in the performance of LUNs.

Enabling Compression

One major feature of the Oracle ZFS Storage Appliance is the ability to enable compression at different levels to reduce the amount of storage consumed, with a trade-off against performance.

There are four levels of compression available, with the default set at no compression. In order of increasing compression, these levels are:

- LZJB
- GZIP-2
- GZIP (aka GZIP-6)
- GZIP-9

Figure 9 shows the database latency for reads and writes, measured in ms.

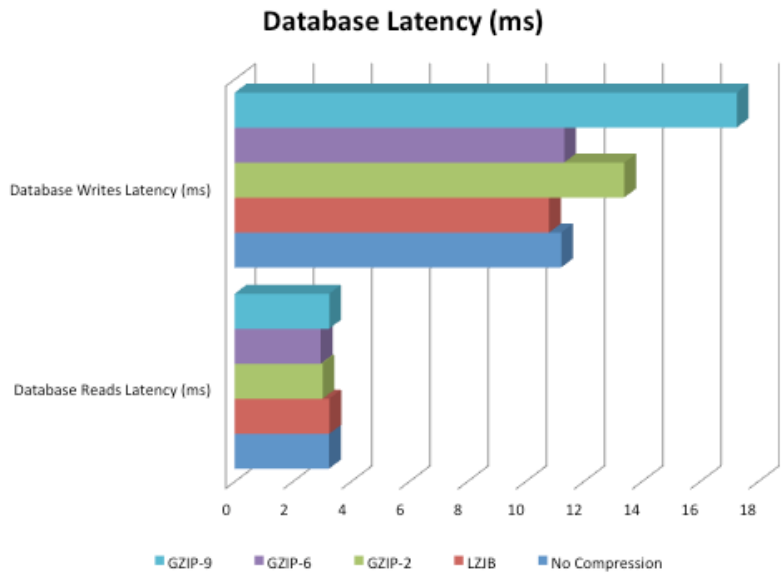

Figure 9. Database IO latency by compression type

The corresponding values for log volumes are shown in Figure 10.

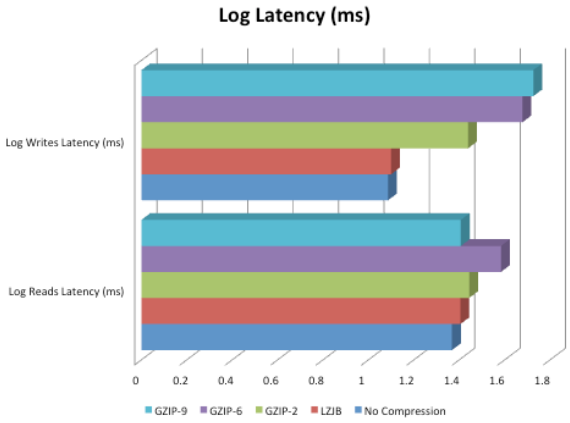

Figure 10. Log IO latency by compression type

As shown in the preceding graphs, for writes, latency increases as the level of compression increases – as expected. However, the variance in reads is not as marked as that for writes for either volume type.

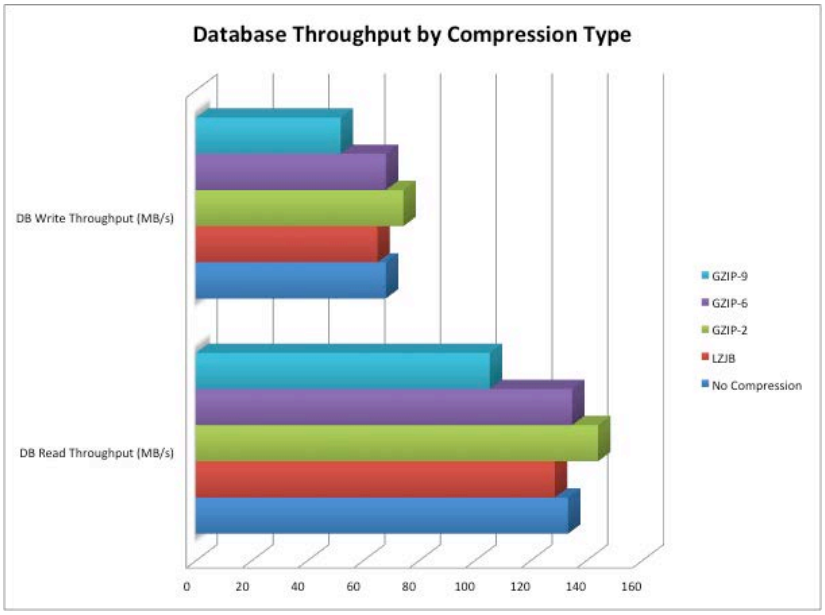

Figure 11. Database throughput by compression type

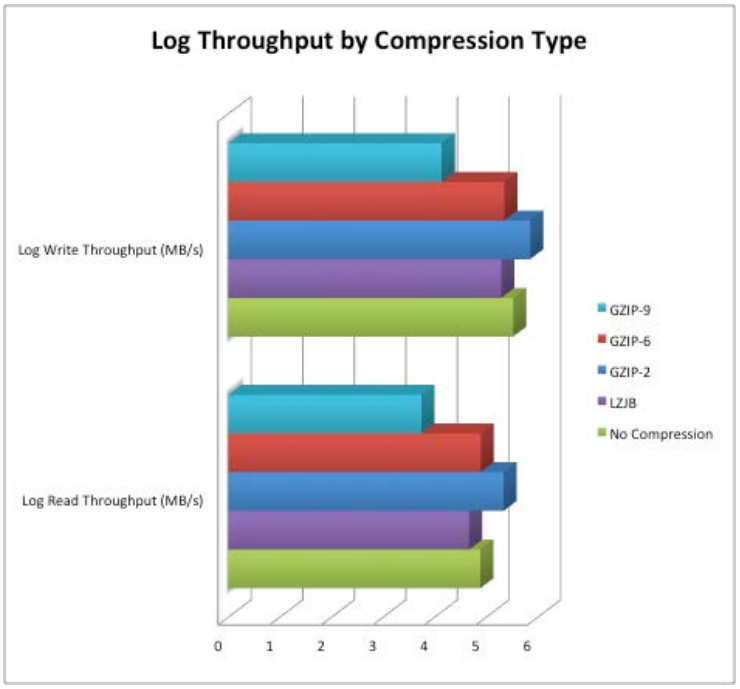

Figure 12. Log throughput by compression level

Based on the preceding figures with the data generated by ESRP, GZIP-2 compression gives the best overall throughput results. The reason for this is the advanced caching and dynamic performance heuristics inherent in the Oracle ZFS Storage Appliance. The CPUs can read the compressed data off disk or from cache, and can decompress faster than simply reading uncompressed data.

However, where latency is important (that is, individual transaction performance rather than the performance of the system as a whole), LZJB compression provides the lowest latency while still allowing a level of compression for both log and database volumes.

The reduction in disk space used varies as compression level increases:

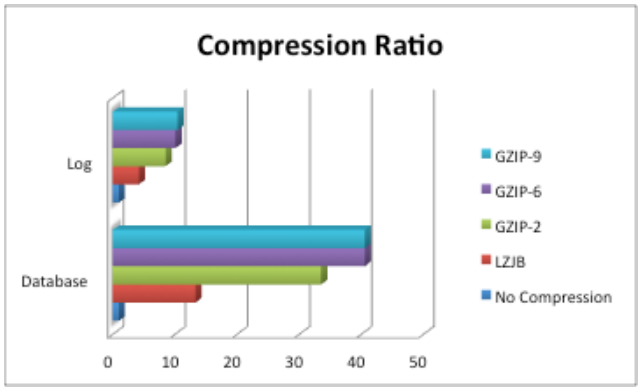

Figure 13. Compression ratios

The actual level of compression returned is dependent on the data type being stored. If the mail system contains a large number of uncompressible images, there may still sufficient value in enabling compression, as the metadata stored by Exchange Server is compressible.

Conclusion

The Oracle ZFS Storage Appliance provides a variety of tuning settings to cater to a number of Microsoft Exchange deployments. The best practice for a Microsoft Exchange Server deployment where the workload of the server is unknown is 64 KB blocksize for both database and log volumes and LZJB compression enabled.

Where the workload is better understood, such as in the case where an existing Microsoft Exchange deployment is being migrated to the Oracle ZFS Storage Appliance, the tables provided for latency and throughput statistics allow a more appropriate blocksize to be chosen. It is also worth remembering that the figures provided in this document are not absolute values but should be considered as relative differences between the blocksizes and/or compression levels.

The official ESRP 2013 tests show that the 128 KB blocksize with LZJB compression enabled is the all-round best compromise, but this also includes backup performance and log replay performance which would not normally affect the perceived performance as experienced by users (except in a 60-24-7-365 Exchange Server environment.)

Adding additional disk shelves with log devices allows the same platform to provide the necessary infrastructure to support the full size range of enterprise-level deployments.

The additional data services provided by the Oracle ZFS Storage Appliance in parallel with its connectivity options (Infiniband, iSCSI, Fibre Channel, SMB, NFS) combine to continue to make the Oracle ZFS Storage Appliance an essential choice for any procurement short list.

References

See the following resources for additional information relating to the products covered in this document.

Microsoft Exchange Server Database Availability Groups
http://technet.microsoft.com/en-gb/library/dd979799(v=exchg.150).aspx

Microsoft Windows Server Utility 'fsutil'
https://technet.microsoft.com/en-us/library/cc753059.aspx

Oracle ZFS Storage Appliance White Papers and Subject-Specific Resources
http://www.oracle.com/technetwork/server-storage/sun-unified-storage/documentation/index.html

Oracle ZFS Storage Appliance Product Information
https://www.oracle.com/storage/nas/index.html

Oracle ZFS Storage Appliance Documentation Library, including Installation, Analytics, Customer Service, and Administration guides:
http://www.oracle.com/technetwork/documentation/oracle-unified-ss-193371.html

The *Oracle ZFS Storage Appliance Administration Guide* is also available through the Oracle ZFS Storage Appliance help context.
The Help function in Oracle ZFS Storage Appliance can be accessed through the browser user interface.

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

ORACLE

CONNECT WITH US

blogs.oracle.com/oracle
facebook.com/oracle
twitter.com/oracle
oracle.com

Integrated Cloud Applications & Platform Services

Performance Tuning the Oracle ZFS Storage Appliance for Microsoft Exchange 2013
March 2016, Version 1.0
Author: Andrew Ness
Oracle Application Integration Engineering